Discord鯖に以下の通知を送れるようにします。
・樹脂溢れ(前後)
・参量物質変化器使用可能(前後)
・洞天宝銭満タン(前後)
・デイリー未受け取り
・探索派遣

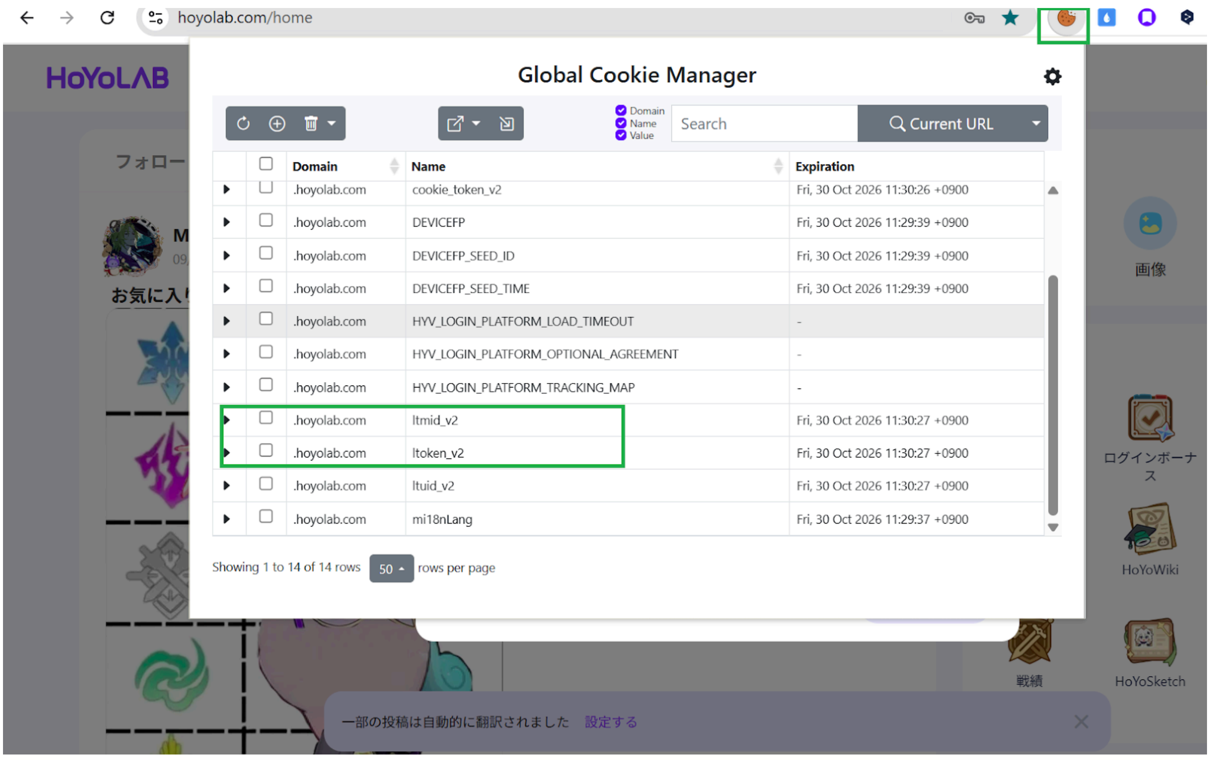
やり方とソースコードです。たぶんそのうち消します。また私は実行により生じた一切の責任を負いません。(API叩くだけなので全く問題ないだろうけどAPI自体消されちゃうかも。)

用意するもの: PC版Google Chromeと拡張機能 Global Cookie Manager (Cookie情報を取得できるものならなんでもいいです、Chromeじゃなくてもいいし拡張機能じゃなくてDeveloper Toolとかでも。)

Google Apps Script (GAS) のトリガーを使って定期実行するようにしてます、AWSとかGCPでもいいけどGASの方が簡単。

手順①
Discordのサーバーを作ってWebhookを追加し、WebhookURLを取得してください(後で使います)。表示される画像と名前(傀儡とか)もここで設定してください。
参考: 【Discord】ウェブフックの作り方と作成できないときの対処方法【ディスコード】 - ゲームウィズ の「ウェブフックURLをコピー」の部分まで。

手順②
ChromeでHoyolabにログインし、右上の拡張機能のメニューからGlobal Cookie Manager を開き、ldtoken_v2 と ltuid_v2 をコピーしてください(後で使います)。



ここまで出来たらほぼ終わり。

手順③

[Google Apps Script](#) を開き、「新しいプロジェクト」を押すと以下の画面になります。



もともとあったコードを削除し、以下のコードを丸ごとコピペしてください(黒いとこ全部。コピーボタンが無くてすみません)。
最初の4行の WebhookUrl, ltokenV2, ltuidV2, UID は各自の値に置き換えてください。

```
const webhookUrl = "Discordから取得したWebhookURL";
const ltokenV2 = "Cookieから取得したltoken_v2";
const ltuidV2 = "Cookieから取得したltuid_v2";
const UID = "あなたの原神UID";

let messageContent = "";

function fetchGenshinDailyNote() {
  const url =

"https://bbs-api-os.hoyoverse.com/game_record/genshin/api/dailyNote";
  const params = {
    role_id: UID,
    server: "os_asia",
    schedule_type: 1,
  };
  const headers = {
    Cookie: `ltoken_v2=${ltokenV2}; ltuid_v2=${ltuidV2}`,
  };

  const options = {
    method: "get",
    headers: headers,
    muteHttpExceptions: true,
  };
```

```
  const queryString = Object.keys(params)
    .map(
      (key) =>
`${encodeURIComponent(key)}=${encodeURIComponent(params[key])}`
    )
    .join("&");

  const response = UrlFetchApp.fetch(`${url}?${queryString}`, options);
  return JSON.parse(response.getContentText());
}

function sendDiscordMessage(content, username = null, avatarUrl = null)
{
  const payload = {
    content: content,
    // username: "カスタムユーザー名", // オプション:ウェブフックのデフォルト名を上
書き
    // avatar_url: "アバター画像のURL" // オプション:ウェブフックのデフォルトアバ
ターを上書き
  };

  const options = {
    method: "post",
    contentType: "application/json",
    payload: JSON.stringify(payload),
    muteHttpExceptions: true,
  };

  const response = UrlFetchApp.fetch(webhookUrl, options);
  if (response.getResponseCode() === 200 || response.getResponseCode()
=== 204) {
    Logger.log("Discordにメッセージが正常に送信されました。");
  } else {
    Logger.log(
      `メッセージの送信に失敗しました: ${response.getResponseCode()}
${response.getContentText()}`
    );
  }
}

function main() {
  try {
    const dailyNote = fetchGenshinDailyNote();

    let shouldSendToDiscord = false; // Discordに送るかどうかを制御する
```

```boolean

    if (
      dailyNote.data.resin_recovery_time > 0 &&
      dailyNote.data.resin_recovery_time < 3600
    ) {
      messageContent += `樹脂があと${Math.ceil(
        dailyNote.data.resin_recovery_time / 60
      )}分で溢れるわよ。とっとと消費しなさい。 ${
        dailyNote.data.current_resin
      }/${dailyNote.data.max_resin}\n`;
      shouldSendToDiscord = true;
    } else if (dailyNote.data.current_resin ===
dailyNote.data.max_resin) {
      messageContent += `樹脂が溢れたわよ。
${dailyNote.data.current_resin}/${dailyNote.data.max_resin}\n`;
      shouldSendToDiscord = true;
    }

    if (dailyNote.data.home_coin_recovery_time < 3600) {
      messageContent += `洞天宝銭があと${Math.ceil(
        dailyNote.data.home_coin_recovery_time / 60
      )}分で溢れるわよ。 ${dailyNote.data.current_home_coin}/${
        dailyNote.data.max_home_coin
      }\n`;
      shouldSendToDiscord = true;
    }

    let { Day, Hour, Minute, Second, reached } =
      dailyNote.data.transformer.recovery_time;
    let recoveryTime =
      Day * 24 * 60 + Hour * 60 + Minute + Math.floor(Second / 60);

    if (!reached && recoveryTime < 60) {
      messageContent += `参量物質変化器はあと${Math.ceil(
        recoveryTime
      )}分で使用可能になるわ。\n`;
      shouldSendToDiscord = true;
    } else if (reached) {
      messageContent += `参量物質変化器が使用可能になったわよ。\n`;
      shouldSendToDiscord = true;
    }

    let expeditionsFinished = dailyNote.data.expeditions.filter(
      (ele) => ele.status === "Finished"
    ).length;
```

```
    if (expeditionsFinished > 0 && new Date().getHours() % 2 === 0) {
      messageContent += `探索派遣が${expeditionsFinished}つ完了しているわ
よ。\n`;
      shouldSendToDiscord = true;
    }

    if (
      dailyNote.data.is_extra_task_reward_received === false &&
      new Date().getHours() >= 21
    ) {
      messageContent += `あら、まだ今日のデイリー依頼が終わってないじゃない。何を
もたもたしているの、とっとと終わらせなさい。\n`;
      shouldSendToDiscord = true;
    }

    if (shouldSendToDiscord) {
      sendDiscordMessage(messageContent);
    } else {
      Logger.log("No important notifications to send.");
    }
  } catch (error) {
    messageContent = "情報の取得に失敗したわ。ログイン状態を見直しなさい。";
    sendDiscordMessage(messageContent);

    Logger.log("Error in main: " + error);
  }
}
```

セリフとか時間の細かい設定はお好みで変えてください。GPTに投げれば大体教えてくれると思います。

あとは左側のメニューからトリガー→トリガーを追加,実行する関数→main、イベントのソースを選択→時間主導型、時間ベースのトリガーのタイプ→時間(or分)ベースのタイマー に変更。時間の間隔はお好みで。



これで定期実行されるようになるはずです。
API自体は公開されてないけどHoyolabアプリで使われてるものだと思います(スマホのウィジェットで残り樹脂とか確認できるやつ)。一日数万リクエストとか極端に多くすると変えられちゃう可能性もあるのでくれぐれも毎秒実行とかはしないでください。

以上です。雑に書いたので分からないことがあったらXで @houjoukaika をメンションしてください、多分答えます。
お疲れさまでした！